Multiscale Shaders for the Efficient Realistic Rendering of Pine-Trees

Alexandre Meyer Fabrice Neyret

iMAGIS[†]-GRAVIR / IMAG-INRIA

Email:{Alexandre.Meyer|Fabrice.Neyret}@imag.fr

Abstract

The frame of our work is the efficient realistic rendering of scenes containing a huge amount of data for which an a priori knowledge is available. In this paper, we present a new model able to render forests of pine-trees efficiently in ray-tracing and free of aliasing. This model is based on three scales of shaders representing the geometry (i.e. needles) that is smaller than a pixel size. These shaders are computed by analytically integrating the illumination reflected by this geometry using the a priori knowledge. They include the effects of local illumination, shadows and opacity within the concerned volume of data.

Key words: Shaders, levels of details, natural scenes, ray-tracing

1 Introduction

Natural scenes such as landscapes and forests are extremely complex in term of the number of geometric primitives that lies in the field of view. Trees belongs to this category of objects that have no defined surfaces, which makes most of the geometry inside the canope potentially visible and potentially enlightened. Ray-tracing such a scene is thus very costly and very subject to aliasing. On the other hand, geometric details like needles or leaves are so small that they usually cannot be seen except for the nearest trees. Boughs of leaves themselves merge with distance. It is thus tempting to replace the indistinguishable data by a fuzzy primitive that would reproduce the same photometric behavior that the group of geometry it represents. In this paper, we propose such primitives at several scales for the particular case of the pine-tree or fir-tree. This approach can certainly be extended to other kind of trees, or to other objects for which an a priori knowledge on the shape distribution exists.

* *iMAGIS* is a joint research project of CNRS/INRIA/UJF/INPG. iMAGIS, INRIA Rhône-Alpes - ZIRST, 655 avenue de l'Europe, 38330 Montbonnot Saint Martin, France.

2 Previous Work

Which aspects matter in the photometric behavior of a group of shapes ? The cumulated local illumination, the cumulated shadows, and the cumulated opacity. An a priori knowledge on the matter distribution will help to compute them. Conversely, the exact shape and location of single parts are unimportant as soon as they introduce no correlation in the visibility of parts that is not already captured in the a priori knowledge. We survey now the existing models which purpose is to represent the effects of the small scales and the rendering models of trees.

Surface shaders

Some primitives have been proposed early to figure small surface details without rendering explicitly their geometry: Blinn has introduced textures of Phong parameters [3] and bump-mapping [4] in this purpose.

Kajiya has introduced the idea of hierarchy of models [11]. In this paper, he suggests to switch from geometry to mapping of Phong parameters, then to reflectance model[1] according to the distance. Transitions from geometry to bump and from bump to reflectance have been proposed in [1, 5, 7].

Several reflectance models based on the surface microgeometry have been developed [11, 24, 8, 17, 6, 10, 9]. Most of these models consist in proposing a representation of the matter distribution, then to integrate the local illumination while addressing the visibility of the details for the viewer and for the light (i.e. self-shadows).

Volume shaders

All the models above are designed for surface details. In the scope of 3D matter distributions, Blinn has early proposed a reflection model for volumes of dust [2] represented by micro-spheres. Stam has developed in [21] a stochastic model which allows the analytical integration of the stochastic distribution of matter to represent details in clouds. Kajiya introduced the volumetric textures [12]

[1] reflectance models are also named *shaders*.

in the scope of fur rendering. A shader (i.e. a local illumination model) is derived to integrate the light reflected on hairs represented by cylinders. This cylinder shader has been improved in [8] and Neyret has extended the volumetric textures representation in [14, 15] by introducing a shader able to integrate at one scale the shaders representing a thinner scale.

Contrary to the models of surface details, most of the 3D models presented above fail to address analytically the visibility of the details from the viewer or from the light. For instance the representation of the 3D microgeometry by a normal distribution in [14] cannot capture the visibility (otherwise the normal distribution should depend on the point of view), whereas the stochastic model of [22] can.

Dedicated tree rendering models

On the other hand, several models dedicated to an efficient representation and rendering of trees and forests have been proposed, using ray-tracing or real-time techniques [16, 23].

Reeves introduced the particles systems [19, 20]. This representation is dedicated to objects made of a huge amount of small long primitives that are drawn as simple strokes, well suitable for modeling trees. In his paper, the shadows are faked using a priori simple laws such as proportionality with depth inside a tree.

Max proposed in [13] a hierarchical representation of trees based on color-depth textures following the natural hierarchy of trees.

Early conclusions

To conclude at that point, we can tell that:

- shaders based on a normal distribution function difficultly account for the shadowing inside the small scale.
- shaders consisting in a sampled BRDF are more accurate, but cannot easily be parameterized.
- shaders consisting in analytical BRDF can be both visibility-compliant and parameterized but are not easy to derive.

Our key idea is that such analytical BRDF can be derived when strong a priori knowledge on the matter distribution is available. We think that the matter in trees is structured enough to offer such a possibility. In this paper we address needle-based trees such as pine-tree or fir-tree, as the a priori knowledge on needles distribution is strong.

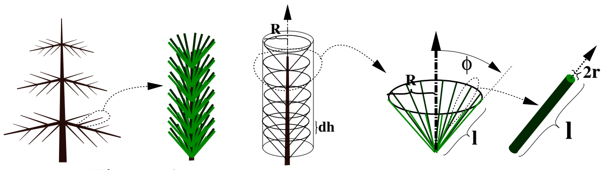

Figure 1: Our hierarchical description of a tree.

3 Contributions

3.1 Our model of pine-tree (see Figure 1)

- A tree is a set of branches and needles that we describe using an L-system [18].
- Branches are classical geometry (i.e. cylinders).
- Needles are cylinders, whose angle ϕ with the branch, length l, radius r, density (i.e. distribution) ρ change slowly along a branch so that they can be considered locally constant.
- The needles layer around the branch (i.e. the bough) is thus a cylinder of radius $R = l\sin(\phi)$.
- We assume that needles are spreaded on cones, with N needles per cone. The distance between cones along a branch is dh. As the gap between two needles end is $\frac{2\pi R}{N}$ and the gap between two cones is dh, it is reasonable to chose $dh = \frac{2.\pi.R}{N} = \sqrt{\rho}$. Whether it is the case or not, we have the relation $dh\frac{2\pi R}{N} = \rho$.

3.2 Multiscale rendering

Depending on the distance, the smallest primitive we consider is either the needle (level 1), the cone (level 2), or the bough (level 3). We render the scene using a simple cone-tracing: the conic ray is used to estimate the apparent size of primitives and to compute their coverage $alpha$ to the pixel. We also use this cones for the shadow rays, assuming point light sources.

The main issue is to compute the global reflectance and opacity of a considered primitive, including the internal shadows. Since we use only conic rays, the rendering is processed with no oversampling at all.

Thus the main contributions of this paper are the multiscale representation that we detail in the next section, the three shaders we derive (detailed in sections 4, 5 and 6), and the method we use to solve the illumination integrals, in particular the geometric interpretation of the visibility and shadows in the level 3 model.

3.3 What we need to compute

In this section we estimate the requirement for the analytical computation of the three shaders. The results and the details of these successive integrations are the object of the three next sections. The $\vec{L}$ and $\vec{V}$ vectors are considered constant because light source and the viewer are far.

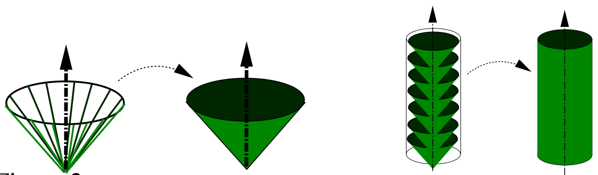

Figure 2: Left*: the continuous cone model.* Right*: the continuous bough model.*

Level 1 (needles)

To shade a needle, we need the amount of diffuse and specular light I_d and I_s reflected by a cylinder [12]. In [17] the integral is correctly expressed in pixel space. We use this form, with different bounds and a cheaper approximation for the specular integral.

We never compute explicitly the intersection of the needles with the cone-ray. We compute instead the intersection of a cone of needles, and we consider the needles that are on the visible part of the cone. Then we sum their illumination.

Level 2 (cones)

We consider that shading a cone of needles is equivalent to shading a continuous semi-opaque cone whose each point reflects the light as a local needle would (see Figure 2 left). The opacity A is the amount of the cone surface covered by needles, so is defined by $A = \frac{2Nr}{\pi R}$. The illumination is A times the integral in pixel space of the cylinder illumination on the visible part of the cone. The front and rear part are considered separately, and only a portion of these parts may be visible in a pixel. This integration is not trivial and requires several approximations.

Level 3 (boughs)

We consider that a bough to be shaded is equivalent to a semi-opaque anisotropic volumetric cylinder made of imbricated cones (see Figure 2 right). The illumination and opacity of front and rear parts of the cones correspond to the level 2 shader already derived (the front part of all the cones are equal, same for the rear parts). The volume model is both continuous and anisotropic: the opacity has to reproduce the same effect as the number of cones traversed by a ray while rendering at level 2, which is strongly dependent of the angle of the ray. The difficult part is the analytical volumetric integration of it, taking into account the visibility and the shadows. Assuming we can use a linear approximation [2] of the opacity composition law, i.e. $(1-A)^n \approx (1-n.A)$, we transpose this integral into a geometric form.

4 Cylinder illumination

We have to integrate the diffuse and specular components into screen space (i.e. we sum the contributions to the pixel color and opacity). Either reflectance or illumination can be derived; one can trivially convert one into the other since we also compute the opacity.

[2]which is valid for $nA \ll 1$, i.e. if the bough is not too dense

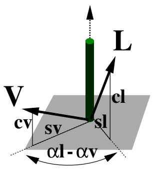

Figure 3: A single needle.

- The diffuse reflectance toward the viewer is

$$R_d^{cyl} = \frac{\int_{cylinder} (N.L)\,\mathbb{I}_{(N.L>0)}\,(N.V)\,\mathbb{I}_{(N.V>0)}\,.dS}{\int_{pixel} (N.V)\,\mathbb{I}_{(N.V>0)}\,.dS}$$

Let c_v and c_l be the projections of V and L on the cylinder axis $\vec{a}$, i.e. $c_v = (\vec{a}.V)$ and $c_l = (\vec{a}.L)$ (see Figure 3).
Let V_p and L_p be the projections of V and L on the plane orthogonal to the cylinder, and s_v and s_l be their norm.

$$R_d^{cyl} = \frac{\int_{\alpha=\alpha_0}^{\alpha_1} s_l \cos(\alpha-\alpha_L) s_v \cos(\alpha-\alpha_V)\,d\alpha}{\int_{\alpha=\alpha_V-\frac{\pi}{2}}^{\alpha_V+\frac{\pi}{2}} s_v \cos(\alpha-\alpha_V)\,d\alpha}$$

with α_V and α_L the angles between a reference in the plane and respectively V_p and L_p. The bounds of visibility α_0 and α_1 are $\alpha_V - \frac{\pi}{2}$ and $\alpha_L + \frac{\pi}{2}$ if $L \times V$ has the same direction than $\vec{a}$. We introduce $\Delta\alpha = |\alpha_V - \alpha_L|$ and then

$$\boxed{R_d^{cyl} = \tfrac{s_l}{4}\left(\sin(\Delta\alpha) + (\pi - \Delta\alpha)\cos(\Delta\alpha)\right)} \tag{1}$$

- The specular reflectance toward the user is

$$R_s^{cyl} = \frac{\int_{cylinder} (N.H)^n\,\mathbb{I}_{(N.H>0)}\,(N.V)\,\mathbb{I}_{(N.V>0)}\,dS}{\int_{pixel} (N.V)\,\mathbb{I}_{(N.V>0)}\,dS}$$

with the half-way vector $H = \frac{V+L}{|V+L|}$ and n the specularity exponent.
Let H_p, c_h, s_h and α_V be defined like for L and V. Then

$$R_s^{cyl} = \frac{\int_{\alpha=\alpha_0}^{\alpha_1} s_h^n \cos^n(\alpha-\alpha_H) s_v \cos(\alpha-\alpha_V)\,d\alpha}{\int_{\alpha=\alpha_V-\frac{\pi}{2}}^{\alpha_V+\frac{\pi}{2}} s_v \cos(\alpha-\alpha_V)\,d\alpha}$$

It is well known that $\cos^n(x)$ is very similar to $e^{-\frac{n}{2}x^2}$ for n large (which is the case). Moreover the density of this function is concentrated on $x = 0$ (the standard deviation is $1/\sqrt{n}$, and n is generally greater than 100), so that $\cos^n(x-x_0)f(x) \approx \cos^n(x-x_0)f(x_0)$
Thus, we have
$R_s^{cyl} \approx (s_h^n s_v \cos(\alpha_H - \alpha_V) \int_{\alpha=\alpha_0}^{\alpha_1} e^{-\frac{n}{2}(\alpha-\alpha_H)^2} d\alpha)/2s_v$
Since $\int_{-\infty}^{\infty} e^{-\frac{1}{2}(\frac{x}{\sigma})^2} = \sqrt{2\pi}\sigma$, the integral above equals

$\sqrt{\frac{2\pi}{n}}$ if $\alpha_H \in [\alpha_0, \alpha_1]$ which is always the case. Thus

$$\boxed{R_s^{cyl} \approx \frac{1}{2} s_h^n \cos(\alpha_H - \alpha_V)\sqrt{\frac{2\pi}{n}}} \quad (2)$$

• The opacity is the proportion of the needle apparent rectangle that falls in the pixel. If the needle is totally covered by the pixel, then

$$\boxed{alpha^{cyl} = \frac{2r.s_v l}{S_{pix}}} \quad (3)$$

where S_{pix} represents the surface of the ray-cone section at the primitive's distance. Thus the diffuse and specular illumination are $I_d = alphaR_d$ and $I_s = alphaR_s$.

5 Cone illumination

As discussed in section 3.3, we consider that the cone is a continuous semi-opaque surface of opacity A, whose each point of the surface reflects the light as a cylinder. Thus, we need to integrate the cylinder illumination into a cone of aperture ϕ for all the valid needle axis positions $\vec{a}_\theta$. In the polar coordinate system associated to the cone, we denote $L = (\theta_L, \phi_L)$, such that ϕ_L is the angle between L and the cone axis. Similarly we denote $V = (\theta_V, \phi_V)$.

• The diffuse illumination is given by:

$$I_d^{cone} = \frac{lA}{4} \int_{\theta=\theta_V-\frac{\pi}{2}}^{\theta_V+\frac{\pi}{2}} s_l s_v \left(\sin(\Delta\alpha) + (\pi - \Delta\alpha)\cos(\Delta\alpha)\right)$$

where ls_v is the apparent length of a needle.

We cannot integrate analytically this formula. As such, we approximate $s_l s_v(\sin(\Delta\alpha) + (\pi - \Delta\alpha)\cos(\Delta\alpha))$ by using the function

$$F = s_l s_v \left(1/2 + \cos(\Delta\alpha)/2\right)\left(2 + (\pi - 2)\cos(\Delta\alpha)\right)$$

which has the same values and derivatives in 0, $\frac{\pi}{2}$ and π and which maximum error is less than 1%.
Since $\cos(\Delta\alpha) = \frac{(L_p.V_p)}{(|L_p|.|V_p|)} = \frac{(L.V)-c_l c_v}{s_l s_v}$ then
$\int F = (L.V + s_l s_v - c_l c_v).(2 + (\pi - 2)(L.V - c_l c_v)/s_l s_v)$

Figure 4: Left: *An example of F curve, for $L = (0, 1.2)$, $V = (1, 1.5)$ and $\phi = .5$. It is very smooth, despite its factors are quite more chaotic.* Right: *the FFT of this curve. Notes that the energy is clearly concentrated on the frequencies 0, 1 and 2, thus the motivation to fit F with a linear combination of 1, $\cos(\theta - \theta_A)$, $\cos(2(\theta - \theta_B))$. NB: the values at the extreme right are the mirroring due to the FFT.*

When tracing this function with Maple for many values of the parameters L, V and ϕ, it appears that the curve is very smooth (Figure 4 left), and looks like a linear combination of 1, $\cos(\theta - \theta_A)$ and $\cos(2(\theta - \theta_B))$. The FFT evaluation on discretized curves shows that there is practically no energy out of the frequencies 0, 1 and 2 (Figure 4 right). As such, we try to fit such a curve to F from the location and value of its extrema. The first factor capture most of the variations of F and is more easy to analyze, so to fit the curve we approximate F by $(L.V) + s_l s_v - c_l c_v$ which seems to have its extrema at the same θ value than F.
The term $c_l c_v - s_l s_v$ equals $\cos(\widehat{AL} + \widehat{AV})$ with $\widehat{AL}$ the angle between the vectors $\vec{a}$ and L , and $\widehat{AV}$ the angle between the vectors $\vec{a}$ and V. These angles vary smoothly between a minimum and a maximum while $\vec{a}$ rotates along the cone, so we model the variation of $\widehat{AL}$ by the form $A_L + B_L\cos(\theta - \theta_L)$ with $A_L = \max(\phi_L, \phi)$, $B_L = \min(\phi_L, \phi)$. We do the same for $\widehat{AV}$.
If we develop $\widehat{AL} + \widehat{AV}$ with this approximation we obtain the expression $A_\Sigma + B_\Sigma\cos(\theta - \theta_\Sigma)$ with
$A_\Sigma = A_L + A_V$, $\quad B_\Sigma^2 = B_L^2 + B_V^2 + 2B_L B_V \cos(\theta_L - \theta_V)$,
$\cos(\theta_\Sigma) = (B_L\cos(\theta_L) + B_V\cos(\theta_V))/B_\Sigma$,
$\sin(\theta_\Sigma) = (B_L\sin(\theta_L) + B_V\sin(\theta_V))/B_\Sigma$

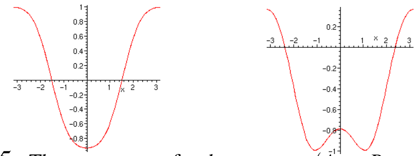

Figure 5: The two aspects for the curve $\cos(A_\Sigma + B_\Sigma * \cos(\theta - \theta_\Sigma))$, *depending whether* $A_\Sigma + B_\Sigma * \cos(\theta - \theta_\Sigma)$ *crosses* π *(right) or not (left).*

We can now search for the extrema of $F \approx (L.V) - \cos(\widehat{AL} + \widehat{AV})$. They correspond either to the extrema of $\widehat{AL} + \widehat{AV}$ or to the location for which $\widehat{AL} + \widehat{AV}$ crosses π. If $\widehat{AL} + \widehat{AV}$ does not cross π, F looks like a cosine function. If it does, F has a hat shape and looks like the combination of a cosine and a cosine at double frequency (see Figure 5). The similarity is high if ϕ_L and ϕ_V are not very close to ϕ. Thus, we can now obtain explicitly the extrema of the curve.
As we are precisely trying to fit F to the form
$(L.V) - (\lambda_0 + \lambda_1\cos(\theta - \theta_m) + \lambda_2\cos(2(\theta - \theta_m)))$
we just have to set the parameters from these extrema:
let $M = \cos(A_\Sigma - B_\Sigma)$ and $m = \cos(A_\Sigma + B_\Sigma)$.
Then $\theta_m = \theta_\Sigma$, $\lambda_1 = \frac{(m-M)}{2}$, $\lambda_0 = \frac{(m+M)}{2} - \lambda_2$, with $\lambda_2 = 0$ if no crossing of π occurs (both $A_\Sigma + B_\Sigma$ and $A_\Sigma - B_\Sigma$ are in $[0, \pi]$),
$\lambda_2 = \frac{\lambda_1 B_\Sigma}{4(2\pi - A_\Sigma)}$ in case of crossing of π ($A_\Sigma + B_\Sigma > \pi > A_\Sigma - B_\Sigma$), Now we can easily obtain the integral of F :

$$\boxed{I_d^{cone} = \frac{l.A}{4}\left(\pi(LV - \lambda_0) - 2\lambda_1\cos(\theta_V - \theta_\Sigma)\right)} \quad (4)$$

where $\cos(\theta_V - \theta_\Sigma) = \frac{B_L\cos(\Delta\theta) + B_V}{\sqrt{B_L^2 + B_V^2 + 2B_L B_V\cos(\Delta\theta)}}$ and $\Delta\theta = \theta_L - \theta_V$.

• The specular illumination is given by

$$I_s^{cone} = \frac{lA}{2}\sqrt{\frac{2\pi}{n}} \int_{\theta=\theta_V-\frac{\pi}{2}}^{\theta_V+\frac{\pi}{2}} s_l s_h^n \cos(\alpha_H - \alpha_V)$$

with ls_v the apparent length of a needle. Once again, s_h^n is a function which density is concentrated on the location where $s_h = 1$, which occurs when $c_h = 0$, i.e. when H is orthogonal to the needle direction $\vec{a}$. Such a location $\theta_H^\perp$ only exists if $\phi_H \in [\frac{\pi}{2} - \phi, \frac{\pi}{2} + \phi]$, otherwise $I_s^{cone} = 0$.

If $\theta_H^\perp$ exists, we have again that $s_h^n f(\theta) \approx s_h^n f(\theta_H^\perp)$.
Since $s_l s_h \cos(\alpha_H - \alpha_V) = (V.H) - c_h c_v$, we finally have

$$\boxed{I_s^{cone} \approx \frac{l.A}{2}\frac{2\pi}{n}(V.H)\varepsilon} \tag{5}$$

where $\varepsilon = 1$ if $\phi_H \in [\frac{\pi}{2} - \phi, \frac{\pi}{2} + \phi]$ otherwise $\varepsilon = 0$.
Note that if both locations where H is orthogonal to $\vec{a}$ occurs on the same face (front or rear), we have that $\varepsilon = 2$.

- The opacity is given by $alpha^{cone} = A \int_{\theta=\theta_V-\frac{\pi}{2}}^{\theta_V+\frac{\pi}{2}} l s_v$

Since $s_v = \sin(\widehat{AV})$, we approximate $\widehat{AV}$ by $A_V + B_V \cos(\theta - \theta_V)$ in the same way that for the diffuse component. That is,

$$\boxed{alpha^{cone} = l.A(\pi\cos(\phi)\cos(\phi_V) - 2\sin(\phi)\sin(\phi_V))} \tag{6}$$

6 Bough illumination

As stated in section 3.3, we consider that the bough is a volume having a cylindrical shape and an anisotropic opacity (as illustrated in Figure 6). We have to proceed to the analytical volume rendering of this cylinder.

Since the opacity A is not constant along the ray and the shadow ray, we have:

$$I = \frac{1}{S_{pix}} \int_{(x,y)\in pixel} \int_{z=near}^{far} A I^{cyl} e^{-\int_0^{l_z}\sigma} e^{-\int_0^{l_{shad}}\sigma} \tag{7}$$

with $e^{-\sigma} = T = (1 - A)$ the anisotropic transparency, l_z the length of the ray within the volume and l_{shad} the length of the shadow ray within the volume.
We need now to explicit the opacity and to do some approximations to make the integral tractable.

6.1 Traversal of a 2D bough

Given an infinite 2D vertical field of parallel needles having a direction ϕ relatively to the top (see Figure 7 left). Let R be the field width, and dh the vertical distance between the needles. A ray in the direction ϕ_r relative to the top crosses the field.
The length of the ray within the field is $R/sin(\phi_r)$
The step between the intersections is $\delta = dh \frac{\sin(\phi)}{\sin|\phi_r - \phi|}$
The average number of intersections is $\frac{R}{dh}\frac{\sin|\phi_r - \phi|}{\sin(\phi)\sin(\phi_r)}$
We denote $k(\phi_r, \phi)$ the quantity $\frac{\sin|\phi_r - \phi|}{\sin(\phi)\sin(\phi_r)} = |\frac{1}{\tan(\phi)} - \frac{1}{\tan(\phi_r)}|$
The opacity of the field along this ray is $1 - T^{\frac{R}{dh}k(\phi_r,\phi)}$
Let denote for short $k_r = k(\phi_r, \phi)$ and $\overline{k}_r = k(\phi_r, \pi - \phi)$
$\overline{k}_r$ corresponds to the traversal of a field which is symmetrical to the first relatively to the vertical.

A 2D bough is composed of two adjacent such fields, the right one with needles of orientation ϕ, and the left one with needles of orientation $\pi - \phi$ (as illustrated on Figure 7 left).
The total number of intersections along a ray is
$\frac{R}{dh}(k_r + \overline{k}_r) = \frac{R}{dh}\frac{\sin|\phi_r-\phi|+\sin|\phi_r+\phi|}{\sin(\phi)\sin(\phi_r)} = \frac{R}{dh}\frac{2}{\tan(\min(\phi,\phi_r))}$
This means that as long as the ray remains outside the cone aperture (i.e. $\phi_r \in [\phi, \pi - \phi]$) the total opacity along the ray is constant, despite it is balanced differently between the front and the rear part. This is true either for a ray or a shadow ray: similarly for the light, in such condition the shadow casted by the bough is constant, while the light enters more easily in one side than in the other. If the ray is inside the cone aperture (above or below), the opacity increases up to 100% for ϕ_r = 0 or π.

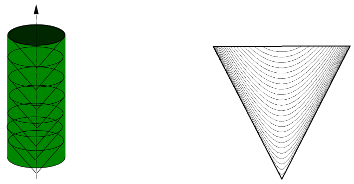

Figure 6: Left : *We model a bough by a semi-opaque volumetric cylinder, which opacity is anisotropic in order to reproduce the variation of the number or intersection between a ray and the sub cones.* Right : *Intersection of the plane P_x with one cone. We approximate the hyberbols by their asymptotes.*

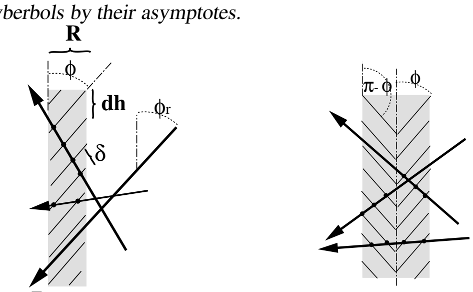

Figure 7: Left: *2D field of parallel 'needles'.* Right: *2D bough. Note the variation of the opacity with the ray direction (mostly on left).*

6.2 Extension to a 3D bough

Let us now come back to our regular bough. In 3D, if a ray crosses the axis of the bough, the situation is equivalent to the 2D situation above. But generally the ray does not cross the axis. Let consider the plane parallel to the cone axis and that contains the ray. Let x be its distance to the axis, thus we name the plane P_x. The intersection of the volume of the bough made of cones with the plane gives a set of hyperboles. We approximate these hyperboles by their two asymptotes (we can see on Figure 6 right that it is reasonable). In that way, the plane contains 'needles' having the same orientation ϕ and offset dh than in 3D, in a field of shrieked thickness $2R_x$ with $R_x = \sqrt{R^2 - x^2}$. So we can compute the number of intersections using the 2D formulas. To estimate the amount of light reaching a point on the ray, we consider a shadow ray starting at that point. Similarly, we introduce the plane parallel to the cone axis and that contains the shadow ray (Figure 8). The number of intersections can be obtained as for the main ray.

6.3 Traversal of a 3D bough

We can now come back to the volumetric integral 7. We choose the (x, y) pixel-surface parameterization so that the $\vec{x}$ axis is orthogonal to the cylinder. Thus x indexes the plane P_x (i.e. x is coherent with the previous section). In consequence we no longer need to integrate along the $\vec{y}$ axis, since the cylinder is homogeneous in this direction. Note that the albedo A in the equation should be corrected to A/δ, since no energy is gathered in the gap between two cones. Similarly on a differential length dl, the opacity is $e^{-\sigma dl} = T^{dl/\delta}$. We proceed to a variable change from (x,z) to (x,z') in the plane orthogonal to the cylinder. This means that we index a point on the ray by its projection on the orthogonal plane. The Jacobian of the transform is $\frac{1}{\sin(\phi)}$. The opacity associated to a differential length dl' on the plane is $T^{\frac{dl'}{\sin(\phi)\delta}} = T^{dl'\frac{k()}{dh}}$

6.4 Splitting the integral into regions

We know from the 2D case that the opacity along the ray is constant on the front half and on the rear half of the traversal (These two halves correspond to the two orientations of the needles in the plane P_x).

The disk has been split into two regions F_V and R_V, the front and the rear relatively to V. On each region $k()$ is constant. In section 5 we have also split the cones into a front face and a rear face, to evaluate the illumination. Let assume that I^{cyl} is constant in each of the two regions of the volume and let approximate it by the mean value I^{cyl}_{front} and I^{cyl}_{rear}. The integral becomes:

$$I = \frac{A}{2Rdh} \int_{x=-R}^{R} \left(k_v I^{cyl}_{front} \int_{z=-R_x}^{0} T^{\frac{k_v}{dh}(R_x+z)} T^{\int_0^{l_{shad}} \frac{k()}{dh}} + \bar{k}_v I^{cyl}_{rear} \int_{z=0}^{Rx} T^{(\frac{k_v}{dh} R_x + \frac{\bar{k}_v}{dh} z)} T^{\int_0^{l_{shad}} \frac{k()}{dh}} \right)$$

In order to get rid of the remaining integral in the exponent, we are now going to split again the disk to separate the front and the rear areas F_L and R_L relatively to L. However the shadow ray length that will appear depends on z on a complicated way, which makes the exponential tricky to integrate analytically.

In order to make the integral tractable, we use the linear approximation of the opacity composition law,

i.e. $(1-A)^n \approx (1-nA)$ which is valid if $nA \ll 1$, i.e. if the bough is not too dense.

Then $(1-A)^{n_1}(1-A)^{n_2} \approx 1 - n_1 A - n_2 A$, which ensures the separation of the factors. Thus the integral is defined as $I = I_{F_V} + I_{R_V} = \frac{A}{2Rdh}\left(I^{cyl}_{front} k_v I_{F'_V} + I^{cyl}_{rear} \bar{k}_v I_{R'_V}\right)$ with

$$I_{F'_V} = \int_{F_V} 1 - A \int_{F_V} \frac{k_v}{dh}(R_x + z) - A \int_{F_V \times R_L} z_{shad} \frac{\bar{k}_l}{dh} - A \int_{F_V \times F_L} z_{shad} \frac{k_l}{dh}$$

$$I_{R'_V} = \int_{R_V} 1 - A \int_{R_V} \left(\frac{k_v}{dh} R_x + \frac{\bar{k}_v}{dh} z\right) - A \int_{R_V \times R_L} z_{shad} \frac{\bar{k}_l}{dh} - A \int_{R_V \times F_L} z_{shad} \frac{k_l}{dh}$$

with $F_V \times R_L$ the region in R_L covered by shadow rays which origin is in F_V, and so on for the other composed regions (see on Figure 8 the representation of these surfaces).

6.5 Geometric integration

We can arrange this as:

$$I_{F'_V} = \pi\frac{R^3}{2} - A\frac{k_v}{dh}\frac{2}{3}R^3 - A\frac{\bar{k}_l}{dh}\int_{F_V \times R_L} z_{shad} - A\frac{k_l}{dh}\int_{F_V \times F_L} z_{shad}$$

$$I_{R'_V} = \pi\frac{R^2}{2} - A\frac{k_v}{dh}\frac{4}{3}R^3 - A\frac{\bar{k}_v}{dh}\frac{2}{3}R^3 - A\frac{\bar{k}_l}{dh}\int_{R_V \times R_L} z_{shad} - A\frac{k_l}{dh}\int_{R_V \times F_L} z_{shad}$$

The four remaining integrals sum the length of the shadow rays starting in each point along the ray and included in the region in subscript, for each ray. Let consider for the moment only the integral along the ray. The shadow-ray sweeps an area while its origin follows the ray. The integral of its length value along the main ray has a strong connection with this surface: it is proportional to it, with a factor $\frac{1}{\sin(l_v)}$ where l_v is the angle between the projections L_P and V_P of L and V in the orthogonal plane. The proof is that if L_P is orthogonal to V_P, then the integral of the length is the regular surface measurement. Otherwise one can come back to this case with a change of variables, which the Jacobian is $\frac{1}{\sin(l_v)}$. So, to compute the integral along the ray, we have to measure the surface of each swept region S_1, S_2, S_3, S_4 using some geometric and trigonometric relations. Then we have to integrate the result for each ray. After some long and unpleasant derivations showing quite complicated formulas in the intermediate stages, we surprisingly found very simple and symmetric results (without any approximation):

$$\int S_1 = (1+\cos(l_v))\frac{R^3}{3}\sin(l_v)$$
$$\int S_2 = (1-\cos(l_v))\frac{R^3}{3}\sin(l_v)$$
$$\int S_3 = (1+\cos(l_v)/3)R^3\sin(l_v)$$
$$\int S_4 = (1-\cos(l_v)/3)R^3\sin(l_v)$$

The $\sin(l_v)$ factors disappear when multiplying by the Jacobian.

6.6 Resulting bough illumination

The opacity is derived trivially:

$1 - \alpha_{F_V} = \frac{1}{2R}\int_{x=-R}^{R} A^{R_x \frac{k_v}{dh}} \approx 1 - \frac{AR}{dh}\frac{\pi}{4}k_v$

i.e. $\boxed{\alpha_{F_V} = ak_v\ ,\ \ \alpha_{R_V} = a\bar{k}_v}$ with $a = \frac{AR}{dh}\frac{\pi}{4}$

We introduce similarly the opacity for the light point of view: $\alpha_{F_L} = ak_l$, $\alpha_{R_L} = a\bar{k}_l$ and finally have $I = I_{F_V} + I_{R_V}$ with

$$\boxed{I_{F_V} = I^{cyl}_{front}\alpha_{F_V}\left(1 - \frac{8}{3\pi^2}\left(2\alpha_{F_v} + (1-\cos(lv))\alpha_{R_L} + (3-\cos(lv))\alpha F_L\right)\right)}$$

$$\boxed{I_{R_V} = I^{cyl}_{rear}\alpha_{R_V}\left(1 - \frac{8}{3\pi^2}\left(4\alpha_{F_v} + 2\alpha_{R_v} + (1+\cos(lv))\alpha_{R_L} + (3+\cos(lv))\alpha F_L\right)\right)}$$

We leave this formula into two separated parts, which allows to render a branch between them.

7 Results

Some resulting images are presented of Figure 9 and Figure 10. We have also compute an animation of the forest scene showing no aliasing artifact.

A major property of our model is the evolution of the cost when the number of needles vary, i.e. the complexity analysis in function of the number N of needles per cone and of the number $\frac{l}{dh}$ of cones on a branch per unit of length (these two numbers are proportional to the square root of the density of needles)[3]. The cost of one shadow ray should evolve the same. However a classical ray-tracer launches a shadow ray for each sample, while for our model the part of the shadow ray that is outside the bough is factorized.

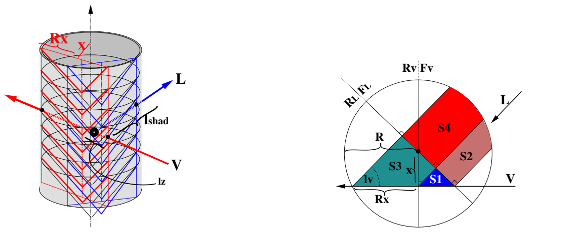

Figure 8: Left: *The volume intersected by the vertical plane containing the ray looks like a 2D bough. Similarly for the shadow ray.* Right: *The volume of the bough cylinder seen in an orthogonal section. The surface of the four regions (see left Figure) $S_1 = F_V \times R_L$, $S_2 = F_V \times F_L$, $S_3 = R_V \times R_L$, $S_4 = R_V \times F_L$ are proportional to the integral of the length of the shadow rays for each possible origin on the ray (only the generic case is figured here). We have to integrate these surfaces for all x.*

We have compared the efficiency to a classical ray-tracer, Rayshade. On Rayshade side, it is important to know that there is a maximum amount of ray per pixel (which is 64), so that when a tree is far (i.e. less than 100 pixels high), Rayshade does not launch enough rays. It might seems efficient, but this is at the price of quality. The fact is that for a image with a lot of high frequencies as image of trees are, the aliasing is not very visible on a single image because it is hard to distinguish noise and information. But the aliasing is obvious during an animation.

The test scene consists of 80 fir-trees that are about 127 pixels high for the closest and 64 for the farthest (Figure 11).

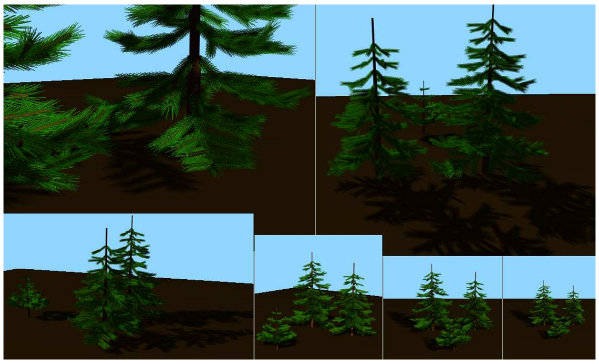

Figure 9: Three fir-trees, from a very close to a far point of view.

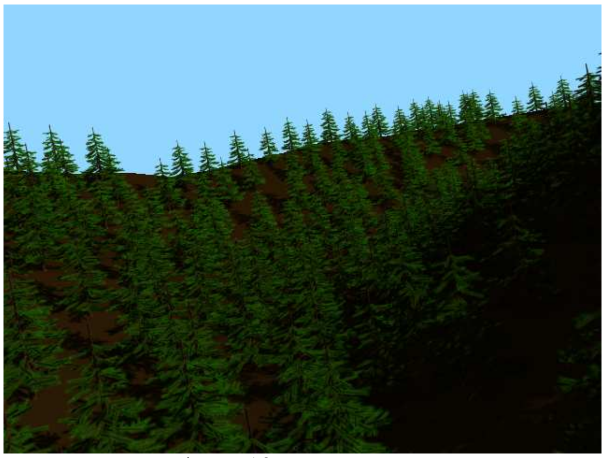

Figure 10: Trees on a hill.

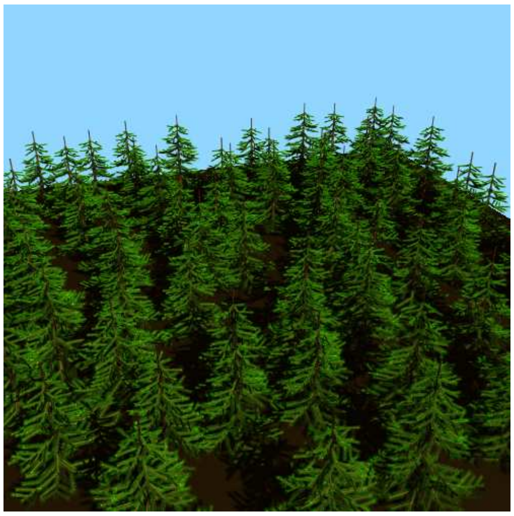

Figure 11: The scene used for the benchmark.

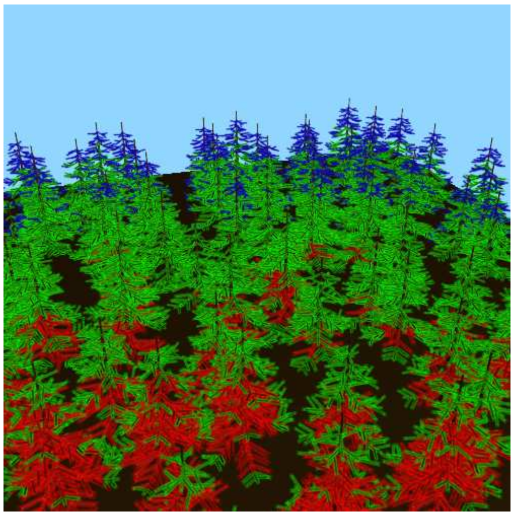

Figure 12: The colors represent the level that is used in our method: red for level 1, green for level 2 and blue for level 3.

[3] if N is multiplied by 2, the number of intersections for level 1 and the number of samples per pixel a ray-tracer should launch are multiplied by 2, while level 2 and level 3 are not affected at all. The same deduction could be done if dh is divided by 2

One fir generally contains 300 branches and about 28700 needles, thus the scene contains about 2 million of needles. Concerning one bough, a cone is 3.94 high, has a radius of 1.6cm, an aperture of $\pi/8$, and the offset between cones is 0.9cm. There are only 12 needles per cone for this tree, whose radius is 0.05cm and length is 4.25cm. On average 4.37 cones are imbricated, so that a ray passing through the axis and orthogonal to the branch would traverse on average 8.75 layers. We run our tests on an SGI Onyx2. The rendering time is 65.3 minutes with Rayshade and 8.1 minutes with our models. Thus our method is about 8 times faster than Rayshade. For landscapes, whose farthest trees are very small, rayshade cannot avoid aliasing due to its 64 samples per pixel limitation. If it could override this limit, the gain would greatly increase in the favor of our method.

8 Conclusion

We have introduced a set of three shaders able to represent at various scales the cumulated effects of the smaller scales without having to sample them, comprising the internal shadows, and taking the visibility into account. As all the required integrations are analytical, this provides at the same time efficiency and image quality (in particular, free of aliasing). However on the theoretical point of view, we would like to improve some of the approximations that have been done. Relaxing the low albedo hypothesis would be interesting either, e.g. using a polynomial law instead of a linear approximation.

The parameters of the shaders allow us to simulate various kind of pine-trees and fir-trees, and to modulate the characteristics inside a single tree (these modulations could be driven in time as well, e.g. to simulate the effects of the wind in a tree). We were able to derive these shaders because the objects we were interested in are very structured. Due to the extended use of the a priori knowledge, these three shaders can simulate nothing but trees made of needles. However, many objects in nature present one kind of structure or another, and even some similarities of structure, so it should be possible for each to derive shaders able to represent analytically each kind. The next step for us will be the simulation of other kind of trees, for which the structure is more stochastic (concerning the distribution and orientation of the leaves). Then it will be also interesting to handle larger scales, exploring larger structures than boughs inside and outside the trees...

Acknowledgments : We wish to thank Celine Loscos and Eugenia Montiel for re-reading this paper. Thanks to Pierre Poulin for discussing during this work.

References

[1] Barry G. Becker and Nelson L. Max. Smooth transitions between bump rendering algorithms. In James T. Kajiya, editor, *Computer Graphics (SIGGRAPH '93 Proceedings)*, volume 27, pages 183–190, August 1993.

[2] J. F. Blinn. Light reflection functions for simulation of clouds and dusty surfaces. In *Computer Graphics (SIGGRAPH '82 Proceedings)*, volume 16(3), pages 21–29, July 1982.

[3] James F. Blinn. Models of light reflection for computer synthesized pictures. In James George, editor, *Computer Graphics (SIGGRAPH '77 Proceedings)*, volume 11(2), pages 192–198, July 1977.

[4] James F. Blinn. Simulation of wrinkled surfaces. In *Computer Graphics (SIGGRAPH '78 Proceedings)*, volume 12(3), pages 286–292, August 1978.

[5] Brian Cabral, Nelson Max, and Rebecca Springmeyer. Bidirectional reflection functions from surface bump maps. In Maureen C. Stone, editor, *Computer Graphics (SIGGRAPH '87 Proceedings)*, volume 21(4), pages 273–281, July 1987.

[6] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics*, 1(1):7–24, January 1982.

[7] Alain Fournier. Normal distribution functions and multiple surfaces. In *Graphics Interface '92 Workshop on Local Illumination*, pages 45–52, May 1992.

[8] Dan B. Goldman. Fake fur rendering. *Proceedings of SIGGRAPH 97*, pages 127–134, August 1997. ISBN 0-89791-896-7. Held in Los Angeles, California.

[9] Jay S. Gondek, Gary W. Meyer, and Jonathan G. Newman. Wavelength dependent reflectance functions. In Andrew Glassner, editor, *Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, Computer Graphics Proceedings, pages 213–220. ACM SIGGRAPH, July 1994.

[10] Xiao D. He, Kenneth E. Torrance, François X. Sillion, and Donald P. Greenberg. A comprehensive physical model for light reflection. *Computer Graphics (Proceedings of SIGGRAPH 91)*, 25(4):175–186, July 1991. ISBN 0-201-56291-X. Held in Las Vegas, Nevada.

[11] James T. Kajiya. Anisotropic reflection models. In B. A. Barsky, editor, *Computer Graphics (SIGGRAPH '85 Proceedings)*, volume 19(3), pages 15–21, July 1985.

[12] James T. Kajiya and Timothy L. Kay. Rendering fur with three dimensional textures. In Jeffrey Lane, editor, *Computer Graphics (SIGGRAPH '89 Proceedings)*, volume 23(3), pages 271–280, July 1989.

[13] Nelson Max. Hierarchical rendering of trees from precomputed multi-layer Z-buffers. In Xavier Pueyo and Peter Schröder, editors, *Eurographics Rendering Workshop 1996*, pages 165–174. Eurographics, Springer Wein, June 1996. ISBN 3-211-82883-4.

[14] Fabrice Neyret. A general and multiscale method for volumetric textures. In *Graphics Interface'95 Proceedings*, pages 83–91, May 1995.

[15] Fabrice Neyret. Modeling animating and rendering complex scenes using volumetric textures. *IEEE Transactions on Visualization and Computer Graphics*, 4(1), January–March 1998. ISSN 1077-2626.

[16] Tsukasa Noma. Bridging between surface rendering and volume rendering for multi-resolution display. In *6th Eurographics Workshop on Rendering*, pages 31–40, June 1995.

[17] Pierre Poulin and Alain Fournier. A model for anisotropic reflection. In Forest Baskett, editor, *Computer Graphics (SIGGRAPH '90 Proceedings)*, volume 24(4), pages 273–282, August 1990.

[18] Przemyslaw Prusinkiewicz, Aristid Lindenmayer, and James Hanan. Developmental models of herbaceous plants for computer imagery purposes. In John Dill, editor, *Computer Graphics (SIGGRAPH '88 Proceedings)*, volume 22, pages 141–150, August 1988.

[19] W. T. Reeves. Particle systems – a technique for modeling a class of fuzzy objects. *ACM Trans. Graphics*, 2:91–108, April 1983.

[20] William T. Reeves and Ricki Blau. Approximate and probabilistic algorithms for shading and rendering structured particle systems. In B. A. Barsky, editor, *Computer Graphics (SIGGRAPH '85 Proceedings)*, volume 19(3), pages 313–322, July 1985.

[21] Jos Stam. Stochastic rendering of density fields. In *Proceedings of Graphics Interface '94*, pages 51–58, Banff, Alberta, Canada, May 1994. Canadian Information Processing Society.

[22] Jos Stam and Eugene Fiume. A multiple-scale stochastic modelling primitive. *Graphics Interface '91*, pages 24–31, June 1991.

[23] Jason Weber and Joseph Penn. Creation and rendering of realistic trees. In Robert Cook, editor, *Computer Graphics (SIGGRAPH '95 Proceedings)*, pages 119–128, August 1995.

[24] Stephen H. Westin, James R. Arvo, and Kenneth E. Torrance. Predicting reflectance functions from complex surfaces. *Computer Graphics (Proceedings of SIGGRAPH 92)*, 26(2):255–264, July 1992. ISBN 0-201-51585-7. Held in Chicago, Illinois.